Assignment #4: T-primes + 贪心

Updated 0337 GMT+8 Oct 15, 2024

2024 fall, Complied by 同学的姓名、院系

说明：

1）请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora https://typoraio.cn ，或者用word）。AC 或者没有AC，都请标上每个题目大致花费时间。

3）课程网站是Canvas平台, https://pku.instructure.com, 学校通知9月19日导入选课名单后启用。**作业写好后，保留在自己手中，待9月20日提交。**

提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、"作业评论"区有上传的md或者doc附件。

4）如果不能在截止前提交作业，请写明原因。

1. 题目

34B. Sale

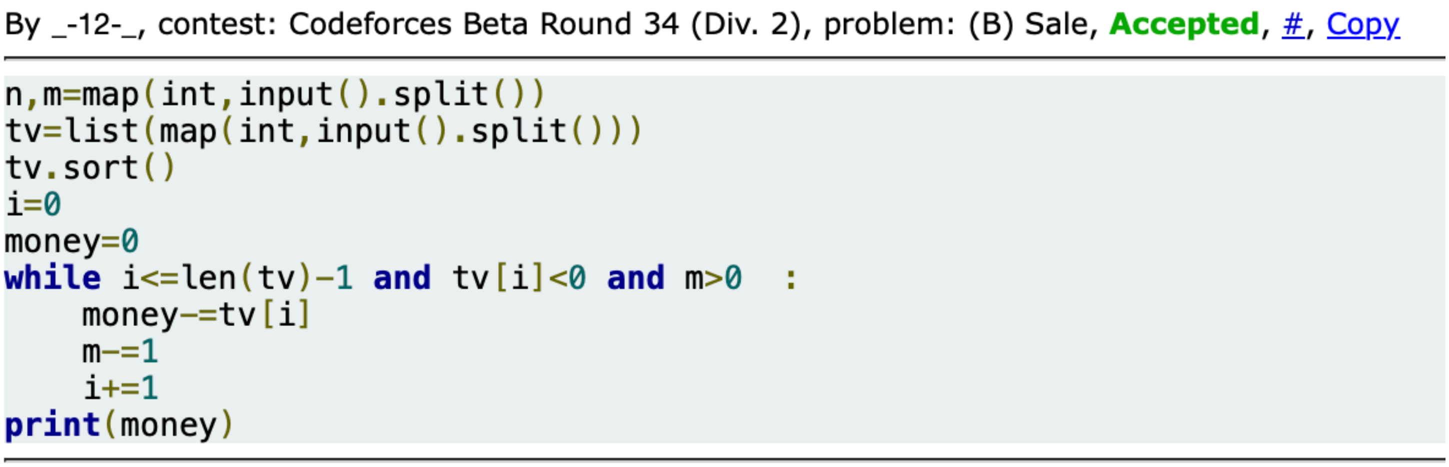
greedy, sorting, 900, https://codeforces.com/problemset/problem/34/B

思路：排序，把价格最负负的电视买掉

代码

```python
# 
n,m=map(int,input().split())
tv=list(map(int,input().split()))
tv.sort()
i=0
money=0
while i<=len(tv)-1 and tv[i]<0 and m>0  :
    money-=tv[i]
    m-=1
    i+=1
print(money)
```

代码运行截图 （至少包含有"Accepted"）</m![20241016132833.png](assets/截屏2024-10-16 13.28.33.png)
ark>



By _-12-_, contest: Codeforces Beta Round 34 (Div. 2), problem: (B) Sale, **Accepted**, #, Copy

```python
n,m=map(int,input().split())
tv=list(map(int,input().split()))
tv.sort()
i=0
money=0
while i<=len(tv)-1 and tv[i]<0 and m>0  :
    money-=tv[i]
    m-=1
    i+=1
print(money)
```

160A. Twins

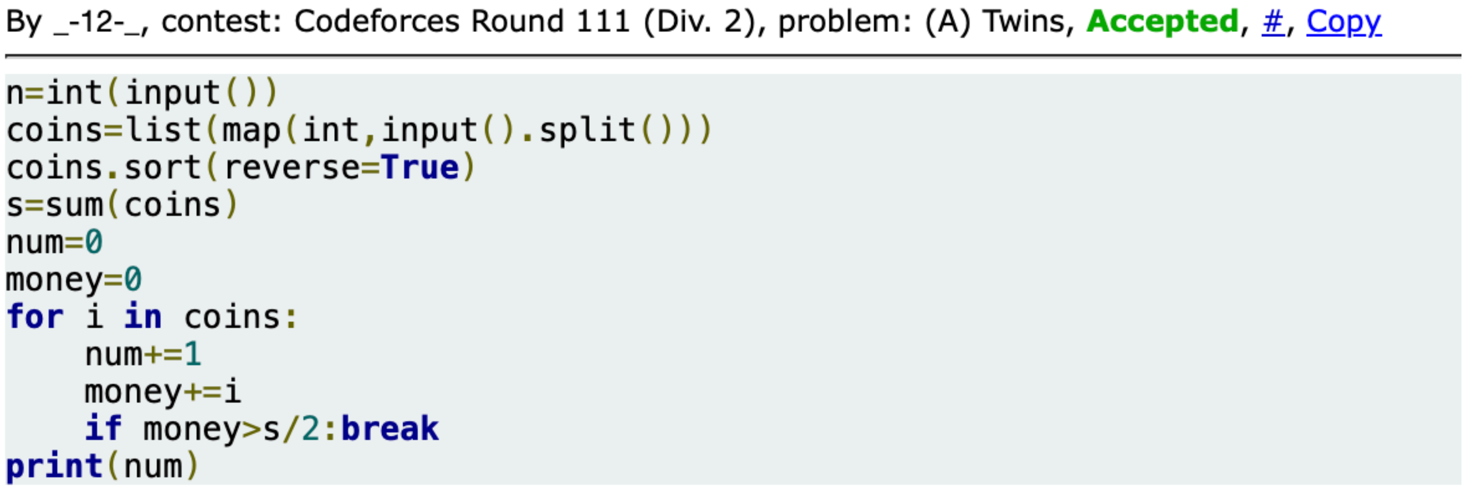
greedy, sortings, 900, https://codeforces.com/problemset/problem/160/A

思路：先拿币值最大的钱

代码

```
n=int(input())
coins=list(map(int,input().split()))
coins.sort(reverse=True)
s=sum(coins)
num=0
money=0
for i in coins:
    num+=1
    money+=i
    if money>s/2:break
print(num)
```

代码运行截图 ==（至少包含有"Accepted"）==



By _-12-_, contest: Codeforces Round 111 (Div. 2), problem: (A) Twins, **Accepted**, #, Copy

```
n=int(input())
coins=list(map(int,input().split()))
coins.sort(reverse=True)
s=sum(coins)
num=0
money=0
for i in coins:
    num+=1
    money+=i
    if money>s/2:break
print(num)
```

1879B. Chips on the Board

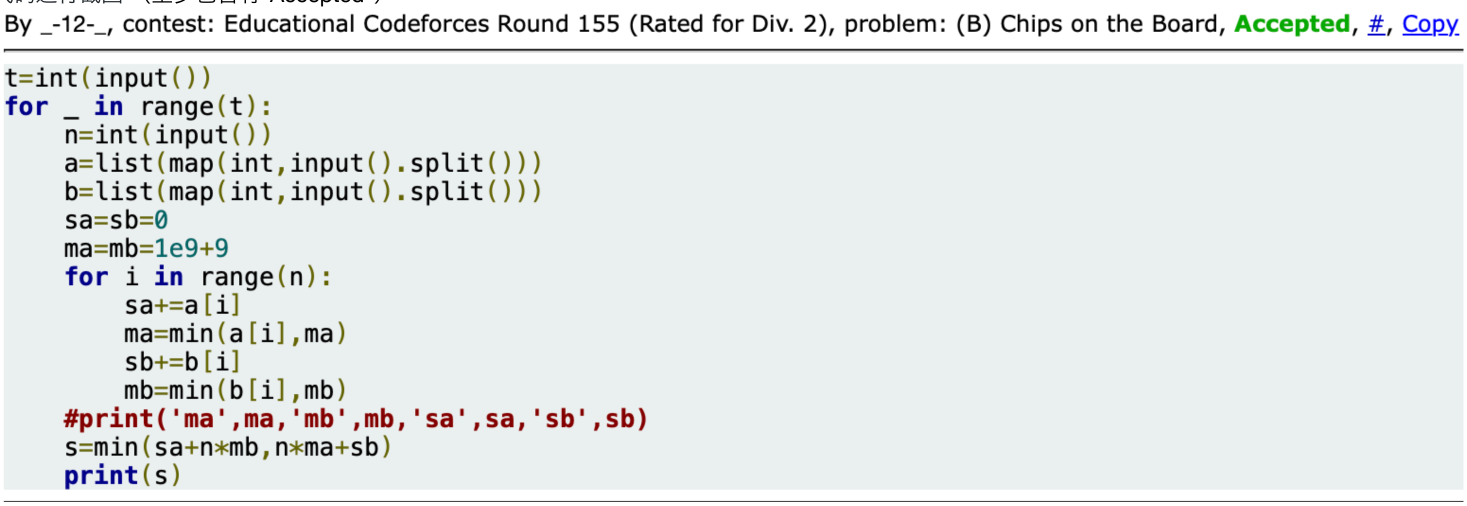
constructive algorithms, greedy, 900, https://codeforces.com/problemset/problem/1879/B

思路：显然应填n个格子，是每一行填一个/每一列填一个，找到一行/一列里面的最小值，每一列/行都填那一行/列

代码

```
t=int(input())
for _ in range(t):
    n=int(input())
    a=list(map(int,input().split()))
    b=list(map(int,input().split()))
    sa=sb=0
    ma=mb=1e9+9
    for i in range(n):
        sa+=a[i]
        ma=min(a[i],ma)
        sb+=b[i]
        mb=min(b[i],mb)
    #print('ma',ma,'mb',mb,'sa',sa,'sb',sb)
    s=min(sa+n*mb,n*ma+sb)
    print(s)
```

代码运行截图 （至少包含有"Accepted"）



By _-12-_, contest: Educational Codeforces Round 155 (Rated for Div. 2), problem: (B) Chips on the Board, **Accepted**, #, Copy

```
t=int(input())
for _ in range(t):
    n=int(input())
    a=list(map(int,input().split()))
    b=list(map(int,input().split()))
    sa=sb=0
    ma=mb=1e9+9
    for i in range(n):
        sa+=a[i]
        ma=min(a[i],ma)
        sb+=b[i]
        mb=min(b[i],mb)
    #print('ma',ma,'mb',mb,'sa',sa,'sb',sb)
    s=min(sa+n*mb,n*ma+sb)
    print(s)
```

158B. Taxi

*special problem, greedy, implementation, 1100, https://codeforces.com/problemset/problem/158/B

思路：装箱问题翻版。用bisect对列表中元素的计数优化了一下,快了100ms。

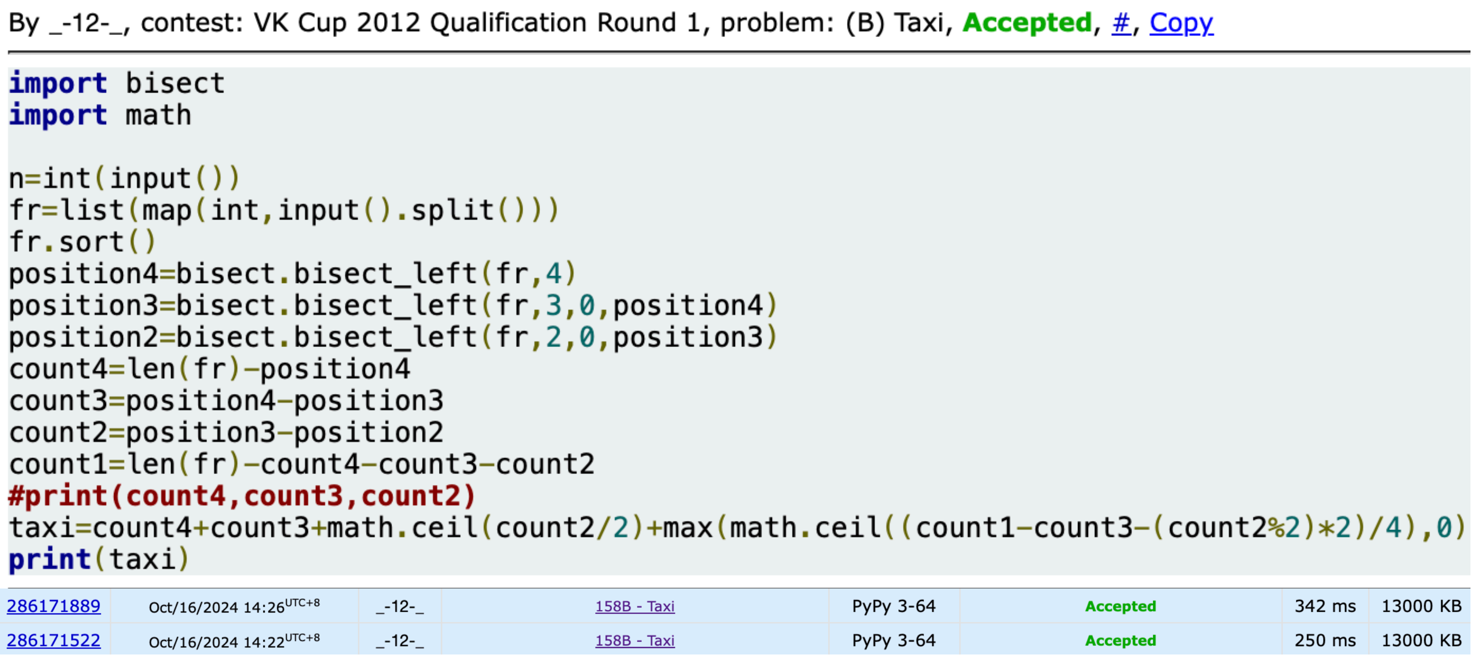
代码

```
import bisect
import math

n=int(input())
fr=list(map(int,input().split()))

fr.sort()
position4=bisect.bisect_left(fr,4)
position3=bisect.bisect_left(fr,3,0,position4)
position2=bisect.bisect_left(fr,2,0,position3)
count4=len(fr)-position4
count3=position4-position3
count2=position3-position2
count1=len(fr)-count4-count3-count2
#print(count4,count3,count2)
taxi=count4+count3+math.ceil(count2/2)+max(math.ceil((count1-count3-(count2%2)*2)/4),0)
'''
#注释里面是去优化的代码
count4=fr.count(4)
count3=fr.count(3)
count2=fr.count(2)
count1=fr.count(1)
'''
taxi=count4+count3+math.ceil(count2/2)+max(math.ceil((count1-count3-(count2%2)*2)/4),0)
print(taxi)
```

代码运行截图 （至少包含有"Accepted"）



By _-12-_, contest: VK Cup 2012 Qualification Round 1, problem: (B) Taxi, **Accepted**, #, Copy

```python
import bisect
import math

n=int(input())
fr=list(map(int,input().split()))
fr.sort()
position4=bisect.bisect_left(fr,4)
position3=bisect.bisect_left(fr,3,0,position4)
position2=bisect.bisect_left(fr,2,0,position3)
count4=len(fr)-position4
count3=position4-position3
count2=position3-position2
count1=len(fr)-count4-count3-count2
#print(count4,count3,count2)
taxi=count4+count3+math.ceil(count2/2)+max(math.ceil((count1-count3-(count2%2)*2)/4),0)
print(taxi)
```

286171889	Oct/16/2024 14:26UTC+8	_-12-_	158B - Taxi	PyPy 3-64	Accepted	342 ms	13000 KB
286171522	Oct/16/2024 14:22UTC+8	_-12-_	158B - Taxi	PyPy 3-64	Accepted	250 ms	13000 KB

*230B. T-primes（选做）

binary search, implementation, math, number theory, 1300, http://codeforces.com/problemset/problem/230/B

思路：集合是个好东西，欧拉筛好东西，不用集合就超时

代码

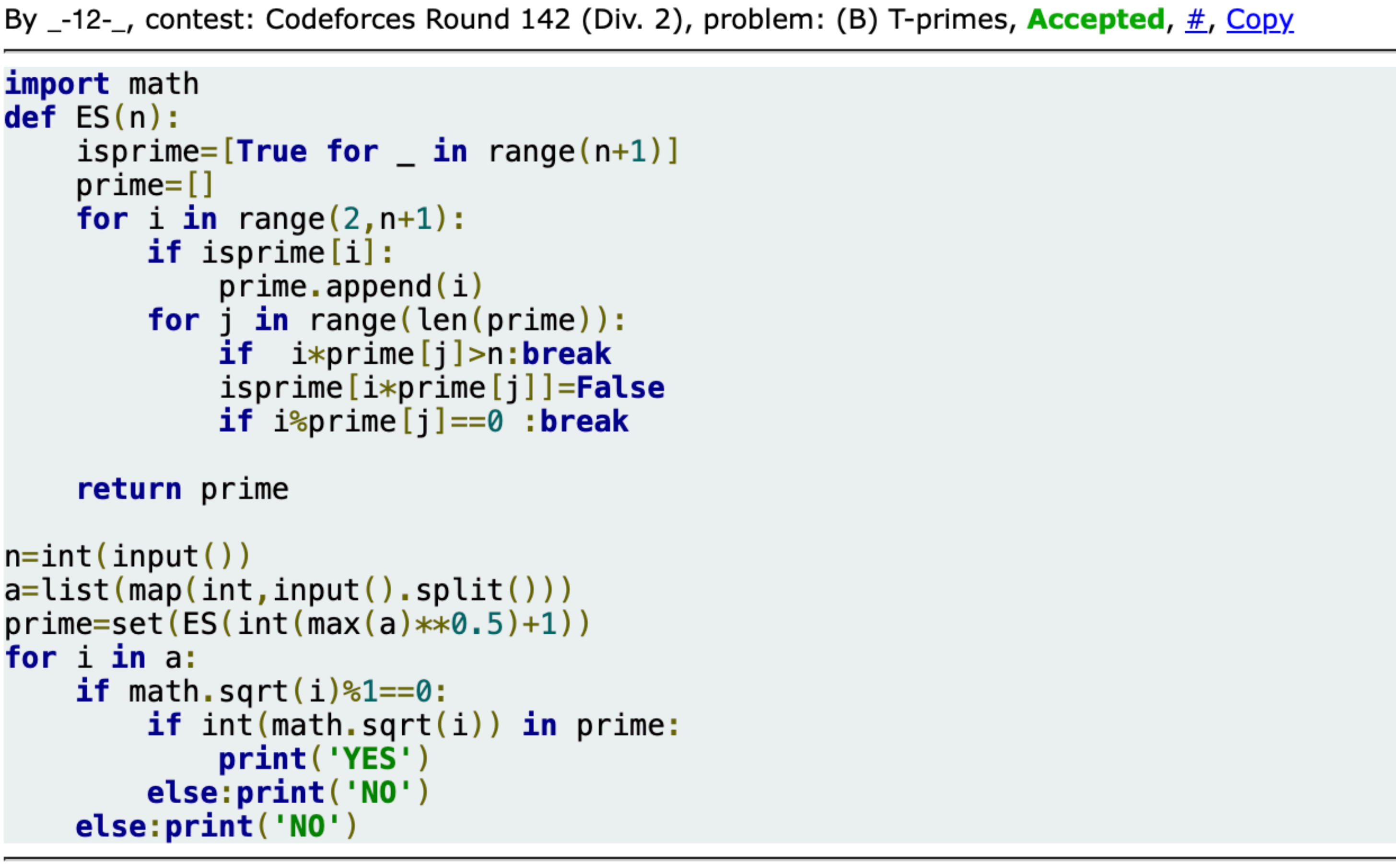
```python
import math
def ES(n):
    isprime=[True for _ in range(n+1)]
    prime=[]
    for i in range(2,n+1):
        if isprime[i]:
            prime.append(i)
        for j in range(len(prime)):
            if  i*prime[j]>n:break
            isprime[i*prime[j]]=False
            if i%prime[j]==0 :break

    return prime

n=int(input())
a=list(map(int,input().split()))
prime=set(ES(int(max(a)**0.5)+1))
for i in a:
    if math.sqrt(i)%1==0:
        if int(math.sqrt(i)) in prime:
            print('YES')
        else:print('NO')
    else:print('NO')

#集合救你命
```

代码运行截图 （至少包含有"Accepted"）



By _-12-_, contest: Codeforces Round 142 (Div. 2), problem: (B) T-primes, Accepted, #, Copy

```python
import math
def ES(n):
    isprime=[True for _ in range(n+1)]
    prime=[]
    for i in range(2,n+1):
        if isprime[i]:
            prime.append(i)
        for j in range(len(prime)):
            if  i*prime[j]>n:break
            isprime[i*prime[j]]=False
            if i%prime[j]==0 :break

    return prime

n=int(input())
a=list(map(int,input().split()))
prime=set(ES(int(max(a)**0.5)+1))
for i in a:
    if math.sqrt(i)%1==0:
        if int(math.sqrt(i)) in prime:
            print('YES')
        else:print('NO')
    else:print('NO')
```

*12559: 最大最小整数 （选做）

greedy, strings, sortings, http://cs101.openjudge.cn/practice/12559

思路：本质上是排序问题，排序的依据是两个字符串a，b，int(a+b)和int(b+a)的大小。然后就是排序算法了。用折半插入超时了，然后特意去学了归并排序（其实就是copy了一下代码...）28msAC了。我做了5天啊啊啊

然后看到数学解法

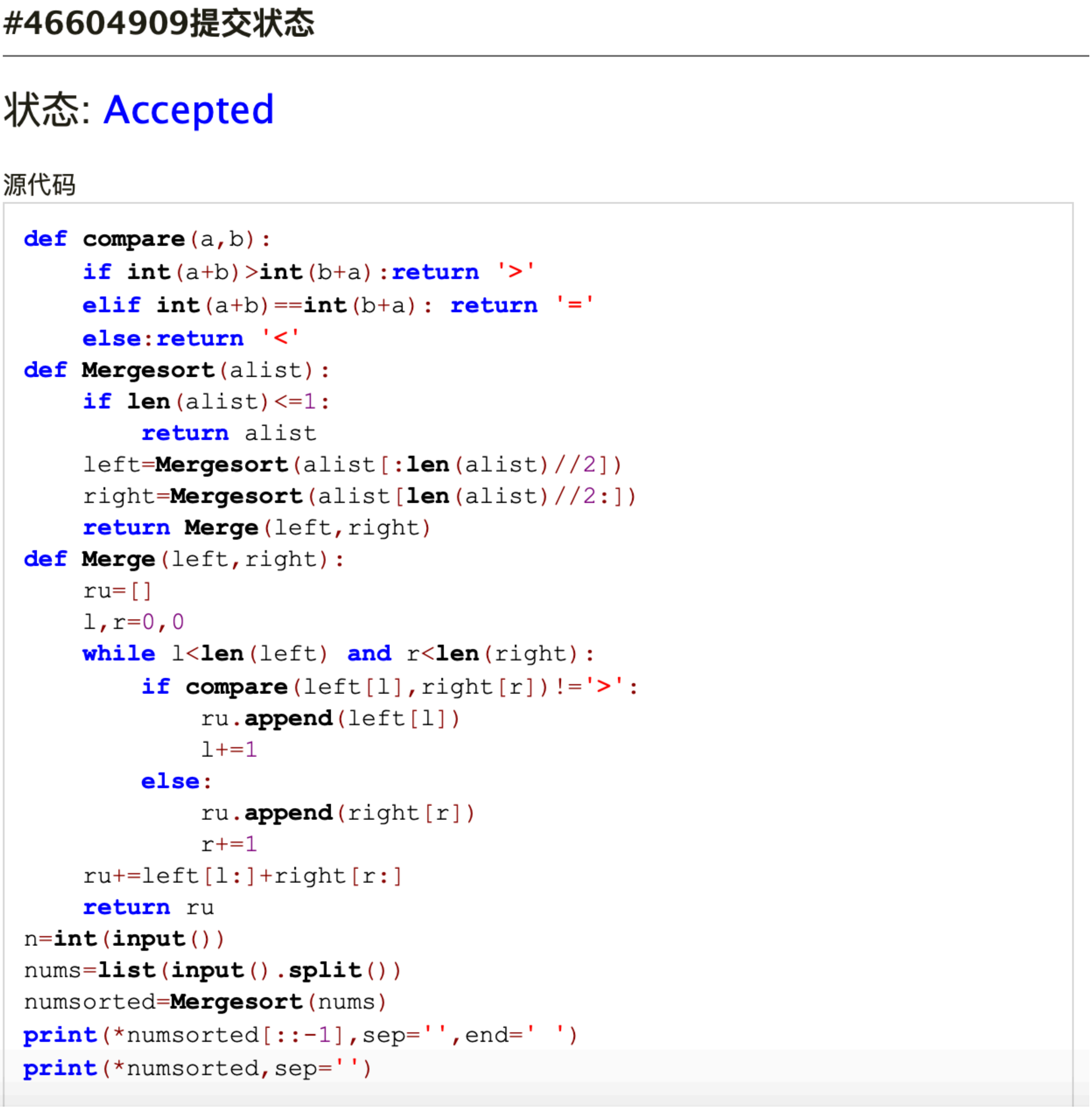
代码220B,22ms

是我智商不够（哭）

代码

```python
def compare(a,b):
    if int(a+b)>int(b+a):return '>'
    elif int(a+b)==int(b+a): return '='
    else:return '<'
def Mergesort(alist):
    if len(alist)<=1:
        return alist
    left=Mergesort(alist[:len(alist)//2])
    right=Mergesort(alist[len(alist)//2:])
    return Merge(left,right)
def Merge(left,right):
    ru=[]
    l,r=0,0
    while l<len(left) and r<len(right):
        if compare(left[l],right[r])!='>':
            ru.append(left[l])
            l+=1
        else:
            ru.append(right[r])
            r+=1
    ru+=left[l:]+right[r:]
    return ru
```

代码运行截图 （至少包含有"Accepted"）



#46604909提交状态

状态: Accepted

源代码

```
def compare(a,b):
    if int(a+b)>int(b+a):return '>'
    elif int(a+b)==int(b+a): return '='
    else:return '<'
def Mergesort(alist):
    if len(alist)<=1:
        return alist
    left=Mergesort(alist[:len(alist)//2])
    right=Mergesort(alist[len(alist)//2:])
    return Merge(left,right)
def Merge(left,right):
    ru=[]
    l,r=0,0
    while l<len(left) and r<len(right):
        if compare(left[l],right[r])!='>':
            ru.append(left[l])
            l+=1
        else:
            ru.append(right[r])
            r+=1
    ru+=left[l:]+right[r:]
    return ru
n=int(input())
nums=list(input().split())
numsorted=Mergesort(nums)
print(*numsorted[::-1],sep='',end=' ')
print(*numsorted,sep='')
```

2. 学习总结和收获

如果作业题目简单，有否额外练习题目，比如：OJ“计概2024fall每日选做”、CF、LeetCode、洛谷等网站题目。

OJ一直在跟，遇到难的会想很久，误打误撞学了归并排序和折半插入排序。准备学递归算法。做不出来的题第二天再看是明智的...